

## Chapter 3

# Memory

Memory is fundamental to both human and artificial intelligence. For humans, it serves as the bedrock of cognition, a vast repository of experiences and knowledge that empowers us to learn, adapt, and navigate the complexities of the world. From infancy, our capacity to encode, store, and retrieve information underpins our ability to acquire language, master skills, and build relationships. Decades of research in neuroscience and cognitive psychology have illuminated the multifaceted role of memory, revealing its influence on our sense of self, creative endeavors, and decision-making processes. Similarly, in the burgeoning field of artificial intelligence, memory is increasingly recognized as a cornerstone of intelligent behavior. Just as humans rely on past experiences to inform present actions, AI agents require robust memory mechanisms to tackle intricate tasks, anticipate future events, and adjust to dynamic environments. Therefore, a deep understanding of human memory – its organization, processes, and limitations – provides invaluable insights for the development of more capable and adaptable AI systems. This section will first provide a concise overview of human memory, focusing on the key stages of encoding, consolidation, and retrieval. We will then transition to exploring the diverse approaches employed in designing AI agent memory systems, ranging from traditional symbolic representations to cutting-edge neural network-based methods. A critical comparison between these artificial memory systems and their human counterparts will highlight existing gaps in areas such as adaptability, contextual understanding, and resilience. Finally, we will consider how principles derived from neuroscience and cognitive psychology can inform future research, suggesting directions that may lead to the creation of artificial memory systems that exhibit greater robustness, nuance, and ultimately, a closer resemblance to the remarkable capabilities of human memory.

### 3.1 Overview of Human Memory

#### 3.1.1 Types of Human Memory

Human memory is often conceptualized as a multi-tiered system that captures, stores, and retrieves information at different levels of processing and timescales. Researchers from the fields of cognitive science, neuroscience, and psychology have proposed various models to describe these levels. A commonly accepted hierarchy distinguishes between sensory memory, short-term memory (including working memory), and long-term memory [170, 171]. Within long-term memory, explicit (declarative) and implicit (non-declarative) forms are further delineated [172]. Figure 3.1 illustrates one such hierarchical framework:

- **Sensory Memory.** Sensory memory is the initial, brief store of raw sensory information. It maintains inputs from the environment for a duration ranging from milliseconds to a few seconds, allowing subsequent processes to determine which portions of the stimulus are relevant for further analysis [173]. Iconic memory (for visual input) [174] and echoic memory (for auditory input) [175] are two well-known subtypes.
- **Short-Term Memory and Working Memory.** Short-term memory (STM) involves holding a limited amount of information in an easily accessible state for seconds to under a minute. The term *working memory* is often used to emphasize the manipulation of that information rather than mere maintenance. While some models treat working memory as a subset of STM, others view it as a distinct system that manages both the storage and active processing of data (for instance, performing arithmetic in one’s head) [176, 177]. The capacity of STM or working memory is finite, typically cited as around seven plus or minus two chunks of information [98], though individual differences and task factors can modulate this figure.

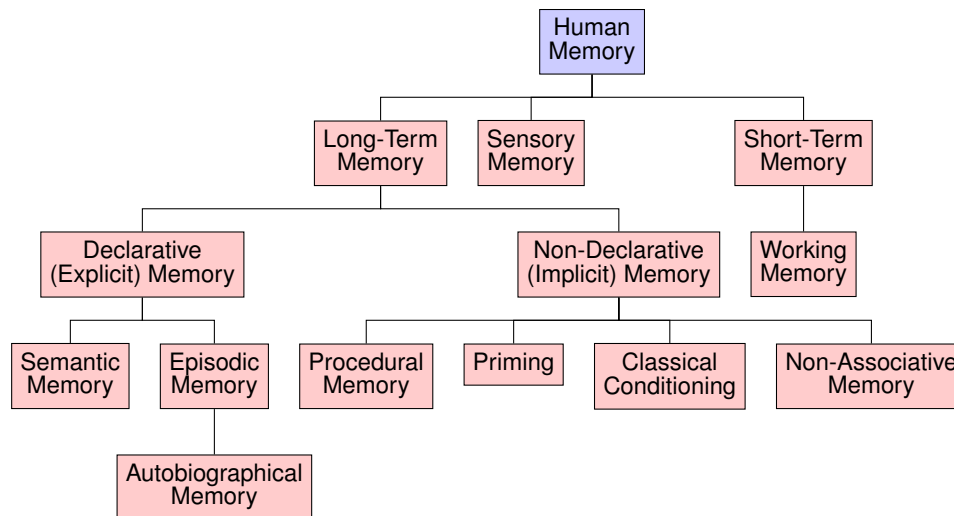


Figure 3.1: The hierarchical taxonomy of human memory system.

- **Long-Term Memory (LTM).** Long-term memory accommodates the more durable storage of information that can persist from hours to decades [178, 179]. This repository supports the learning of skills, the acquisition of factual knowledge, and the recollection of personal experiences. Although long-term memory is sometimes described as having a vast or near-unlimited capacity, factors such as decay, interference, and retrieval cues influence the extent to which stored information can be accessed [180].
  - **Declarative (Explicit) Memory.** Declarative memory encompasses memories that can be consciously recalled and articulated [181]. Within this broad category, researchers often discuss:
    - \* **Semantic Memory:** Factual knowledge about the world, including concepts, words, and their relationships [182]. Examples include recalling the meaning of vocabulary terms or knowing the capital city of a country.
    - \* **Episodic Memory:** Personally experienced events that retain contextual details such as time, place, and the people involved [183]. This form of memory allows individuals to mentally travel back in time to relive past experiences.
    - \* **Autobiographical Memory:** A form of episodic memory focusing on events and experiences related to one’s personal history [184]. While sometimes treated as a sub-category of episodic memory, autobiographical memory places particular emphasis on the self and its evolving life narrative.
  - **Non-Declarative (Implicit) Memory.** Non-declarative memory refers to memories that influence behavior without the need for conscious awareness [185]. Key subtypes include:
    - \* **Procedural Memory:** The gradual acquisition of motor skills and habits (e.g., riding a bicycle, typing on a keyboard) that become automatic with repetition [186, 187].
    - \* **Priming:** The phenomenon in which prior exposure to a stimulus influences subsequent responses, often without explicit recognition of the previous encounter [188].
    - \* **Classical Conditioning:** The learned association between two stimuli, where one stimulus comes to elicit a response originally produced by the other [189].
    - \* **Non-Associative Memory:** Adaptive modifications in behavior following repeated exposure to a single stimulus. Habituation (reduced response to a repeated, harmless stimulus) and sensitization (increased response after exposure to a noxious or intense stimulus) are representative examples [190, 191].

Despite the orderly appearance of these categories, human memory processes often overlap. For example, autobiographical memory is typically nested within episodic memory, yet its particular focus on self-relevant experiences leads some theorists to treat it as a slightly different category. Similarly, the boundary between short-term and working memory can differ depending on the theoretical perspective. Some scholars prefer a more functional, process-oriented view of working memory, while others employ a strictly capacity-oriented concept of short-term storage. In each case, these different perspectives on memory highlight the complexity and nuance of human cognition.

### 3.1.2 Models of Human Memory

Human memory has inspired a wide range of theoretical models, each offering different insights into how information is acquired, organized, and retrieved. Although no single framework commands universal agreement, several influential perspectives have shaped the discourse in cognitive science, neuropsychology, and AI research. The following content highlights some of the most prominent models and architectures used to explain memory’s multiple facets.

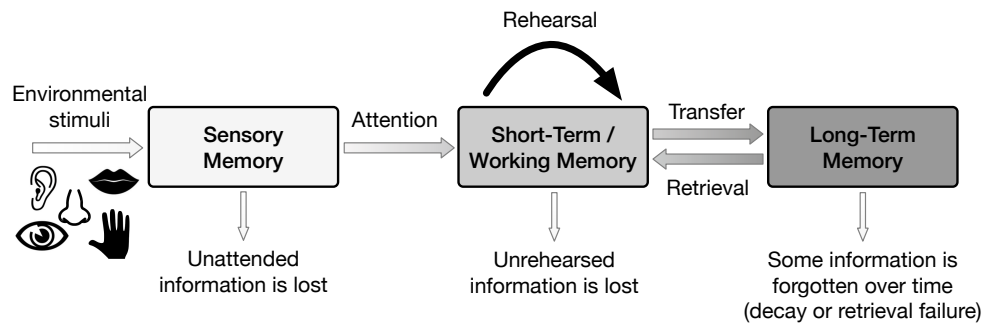


Figure 3.2: Atkinson-Shiffrin three-stage model of human memory [170].

**The Multi-Store (Modal) Model.** A seminal proposal by Atkinson and Shiffrin [170] introduced the multi-store or “modal” model, which posits three main stores for incoming information: *sensory memory*, *short-term memory*, and *long-term memory*. Control processes (e.g., attention, rehearsal) regulate how data transitions across these stores. Figure 3.2 illustrates this model of memory. Despite its relative simplicity, this model remains foundational for understanding how fleeting sensory impressions eventually form stable, long-lasting representations.

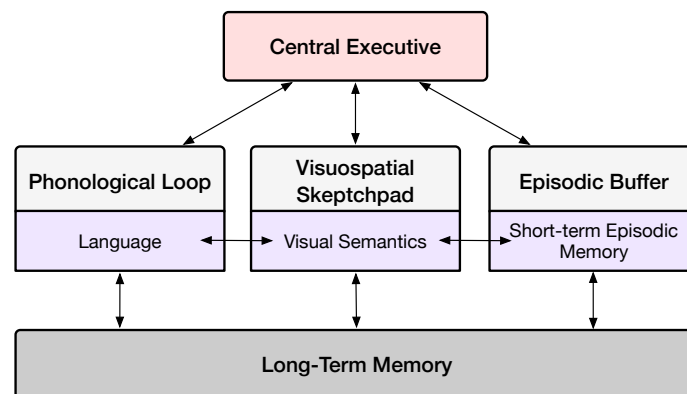


Figure 3.3: Baddeley’s model of working memory [192].

**Working Memory Models.** Recognizing that short-term memory also involves active maintenance, Baddeley and Hitch [192] proposed a *working memory* framework emphasizing the dynamic manipulation of information. Their original model described a central executive that coordinates two subsystems: the phonological loop (verbal) and the visuospatial sketchpad (visual/spatial). A subsequent refinement introduced the episodic buffer to integrate material from these subsystems with long-term memory [193]. Figure 3.3 shows the framework of the working memory model. Alternatives such as Cowan’s embedded-processes model [194] similarly underscore the role of attention in governing how information is briefly sustained and manipulated.

**Serial-Parallel-Independent (SPI) Model.** Initial distinctions between episodic, semantic, and procedural memory were championed by Tulving [195], who later refined his ideas into the Serial-Parallel-Independent (SPI) model, as shown in Figure 3.4. In this framework, memory is divided into two overarching systems. The *cognitive representation system* handles perceptual input and semantic processes, encompassing facts, concepts, and contextual (episodic) knowledge. The *action system*, by contrast, underpins procedural skills such as dance routines, driving maneuvers, or typing proficiency. Tulving’s SPI model posits that memory formation can occur at multiple levels: strictly perceptual encoding can support rudimentary episodic memories, while richer episodic representations benefit from semantic

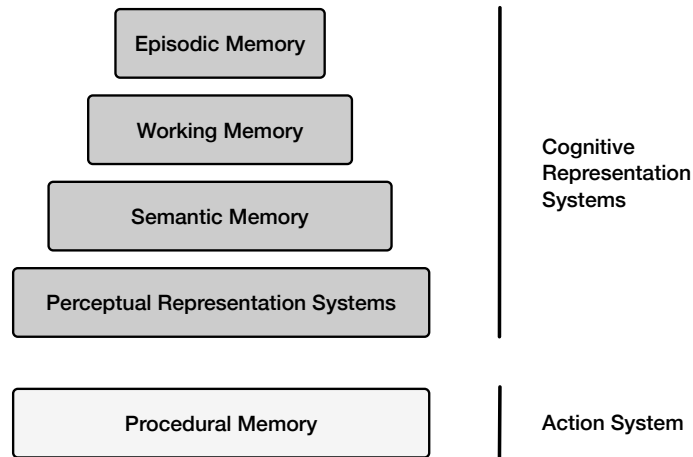


Figure 3.4: The Serial-Parallel Independent (SPI) model of human memory [195].

mediation. For instance, patients with semantic dementia, who struggle to retain word meanings, can still form some episodic memories but often lack the full contextual detail conferred by intact semantic networks. By highlighting the role of procedural memory and its automatic, intuitive nature, the SPI model aims to integrate structure (the content of memory) and function (how memory is used), surpassing earlier accounts that largely focused on explicit storage and retrieval. Despite these strengths, critics note that the model under-specifies how working memory operates within the broader system, and the feedback mechanisms connecting cognitive and action subsystems remain loosely defined.

**Global Workspace Theory (GWT) and the IDA/LIDA Framework.** Global Workspace Theory (GWT), developed by Baars [196], conceptualizes consciousness and working memory as a “broadcast” mechanism that distributes information to specialized processors. Building on GWT, Franklin [197, 198] proposed the *IDA (Intelligent Distribution Agent)* model, later extended to *LIDA (Learning IDA)*, as a comprehensive cognitive architecture. In these frameworks, multiple memory systems (e.g., perceptual, episodic, procedural) interact through iterative “cognitive cycles”, with a global workspace functioning as a hub for attention and decision-making. From an AI standpoint, IDA/LIDA demonstrates how human-like memory processes can be operationalized to guide an agent’s perception, action selection, and learning.

**ACT-R and Cognitive Architectures.** ACT-R (Adaptive Control of Thought—Rational) [199] is a comprehensive cognitive architecture that integrates memory, perception, and motor processes into a unified theoretical framework. It has been applied extensively across diverse domains, including learning and memory, problem-solving, decision-making, language comprehension, perception and attention, cognitive development, and individual differences. Figure 3.5 illustrates the processes of ACT-R. At the core of ACT-R are distinct *modules* (e.g., visual, manual, declarative, procedural) that interact with the system through dedicated *buffers*. Declarative memory stores factual “chunks,” while procedural memory encodes if–then production rules for actions and strategies. Cognition unfolds via a *pattern matcher* that selects a single production to fire based on the current buffer contents. This symbolic production system is augmented by subsymbolic processes, guided by mathematical equations that dynamically regulate activations, retrieval latencies, and production utilities. By combining symbolic and subsymbolic levels, ACT-R provides a mechanistic account of how individuals acquire, retrieve, and apply knowledge—thus shedding light on empirical phenomena such as reaction times, error patterns, and the shaping of learning over time.

Each of the aforementioned models illuminates different aspects of memory. The multi-store model provides a straightforward introduction to storage stages, working memory models emphasize active maintenance and manipulation, and frameworks such as IDA/LIDA or ACT-R embed memory within a comprehensive view of cognition. In practice, researchers often draw upon multiple perspectives, reflecting the intricate nature of human memory and its integral role in perception, learning, and adaptive behavior.

### 3.2 From Human Memory to Agent Memory

Having established the fundamentals of human memory, we now focus on how Large Language Model (LLM)-based agents manage and store information. Memory is not merely a storage mechanism but is fundamental to human and

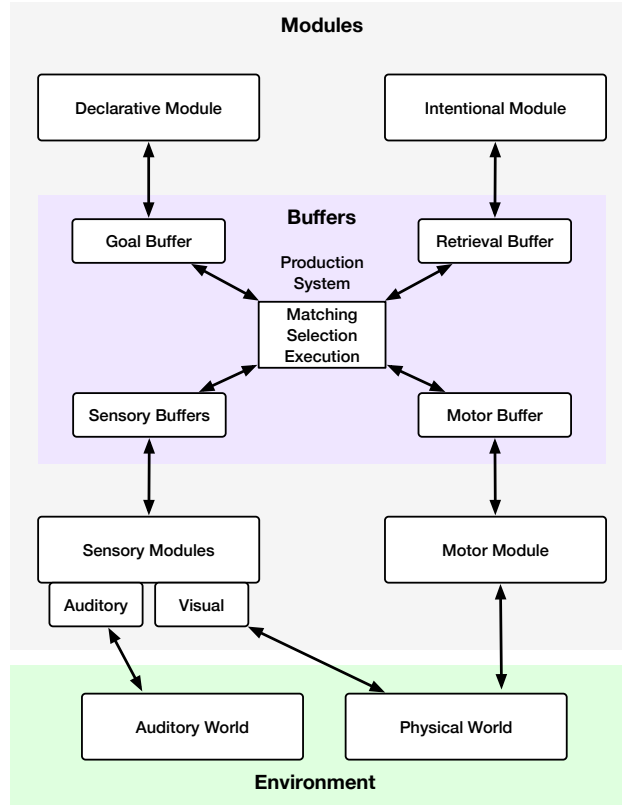


Figure 3.5: An abstraction of the most important processes in the ACT-R model [199].

artificial intelligence. Memory underpins cognition, enabling learning, adaptation, and complex problem-solving for humans. Similarly, for LLM-based agents, memory provides the crucial scaffolding for maintaining context, learning from experience, and acting coherently over time. Without memory, even a highly capable LLM would struggle to adapt to changing circumstances or maintain focus during extended interactions.

While LLM-based agents and biological systems differ fundamentally, the principles guiding human memory—context retention, selective forgetting, and structured retrieval—are highly relevant to agent design. Therefore, examining the parallels and distinctions between human and artificial memory is beneficial. Functionally, we can draw analogies: an agent’s short-term memory buffer resembles the prefrontal cortex’s role in working memory, while long-term storage in a vector database is akin to the hippocampus’s function in consolidating episodic memories. Agent memory design can benefit from emulating human memory’s mechanisms, including selective attention, prioritized encoding, and cue-dependent retrieval. However, crucial differences exist.

Human memory, built upon biological neural networks, integrates storage and computation within neurons’ connections and activity patterns. This offers a high degree of parallelism and adaptability. In contrast, current agent memory systems predominantly rely on digital storage and algorithms, using symbolic representations and logical operations, thus separating storage and computation. This impacts information processing: human memory is associative and dynamic, capable of fuzzy matching and creative leaps, while current agent memory relies on precise matching and vector similarity, struggling with ambiguity. Although digital storage capacity is vast, it cannot yet replicate the complexity and dynamism of human memory, particularly in nuanced pattern recognition and long-term stability. Human memory, while imperfect, excels at extracting crucial information from noisy data. Agent memory systems, in their current stage, are still nascent compared to the intricacies of human memory, facing limitations in organization, integration, adaptive forgetting, and knowledge transfer.

The need for a dedicated memory module in LLM-based agents is paramount. While external knowledge bases (databases, search engines, APIs) [200] provide valuable information, they do not capture the agent’s internal reasoning, partial inferences, or task-specific context. An agentic memory system internalizes interim steps, evolving objectives, and historical dialogue, enabling self-referential exploration and adaptation. This is crucial for tasks requiring the agent to build upon prior judgments or maintain a personalized understanding of user goals.

Early approaches to agent memory, such as appending conversation history to the input prompt (a rudimentary form of working memory) [201], have evolved. Modern architectures employ more sophisticated techniques, including vector embeddings for rapidly retrieving memories [202] and selective incorporation of reasoning chains into subsequent inference steps [203, 204]. These diverse methods share the common goal of managing a large information reservoir without compromising system responsiveness.

However, compared to the sophistication of human memory, current agentic methods have limitations. Many systems lack coherent strategies for long-term memory consolidation, leading to cluttered logs or abrupt information loss. The flexible, bidirectional interplay between stored knowledge and ongoing processing, characteristic of human working memory, is often absent. Metacognitive oversight—selective recall, forgetting, and vigilance against outdated information—is also underdeveloped in LLM-based agents. Balancing comprehensive recall with practical efficiency, as humans do, remains a key challenge.

Building robust and adaptable memory for LLM-based agents involves addressing three core research questions: First, how should memory be represented to capture diverse information types and facilitate efficient access? Second, how can agent memory evolve, incorporating new experiences, adapting to changing contexts, and maintaining consistency? Finally, how can the stored memories effectively enhance reasoning, decision-making, and overall agent performance? The following sections delve into these crucial areas, exploring current approaches, limitations, and potential future directions.

### 3.3 Representation of Agent Memory

Inspired by human cognitive systems [285], current memory architecture in intelligent agents adopts a hierarchical framework that integrates perception through sensory memory [205], real-time decision-making via short-term memory [286, 287], and sustained knowledge retention through long-term memory [288, 289, 48]. This multi-layered structure equips agents to manage immediate tasks while maintaining a broader contextual understanding, fostering adaptability and seamless continuity across diverse interactions.

Specifically, the memory system transforms raw environmental inputs into structured, actionable representations. Sensory memory acts as the gateway, capturing and selectively filtering perceptual signals to provide a foundation for cognitive processing. Short-term memory bridges these immediate perceptions with task-level understanding, buffering recent interactions and enabling dynamic adaptation through experience replay and state management. Long-term memory then consolidates and stores information over extended periods, facilitating cross-task generalization and the accumulation of enduring knowledge.

Together, these memory components form a cohesive cycle of perception, interpretation, and response. This cycle supports real-time decision-making and enables agents to learn and evolve continuously, reflecting an intricate balance between responsiveness and growth. The following delves into the formulation of each memory type, exploring their unique roles and interactions within the agent’s cognitive architecture.

#### 3.3.1 Sensory Memory

In human cognitive systems, sensory memory serves as a mechanism for collecting information through the senses—touch, hearing, vision, and others—and is characterized by its extremely brief lifespan. Analogously, sensory memory functions as the embedded representation of inputs such as text, images, and other perceptual data in intelligent agents. It represents the initial phase of environmental information processing, acting as a gateway for transforming raw observations into meaningful representations for further cognitive processing.

Sensory memory in intelligent agents transcends passive information reception. It dynamically encodes and filters perceptual signals, bridging immediate sensory inputs with the agent’s internal state, objectives, and prior knowledge. This adaptive process facilitates rapid perception of environmental changes, task continuity, and real-time context-aware information processing. Sophisticated attention mechanisms are employed to ensure relevance and focus in the sensory memory layer, forming a critical foundation for decision-making and adaptation.

Formally, sensory memory formation consists of three sequential steps: *perceptual encoding*, *attentional selection*, and *transient retention*. First, perceptual encoding transforms raw sensory signals into processable representations, mathematically expressed as:

$$\phi(o_t) = \text{Encode}(o_t, s_t) \quad (3.1)$$

where  $o_t$  is the sensory input at time  $t$ , and  $s_t$  represents the agent’s state. For instance, RecAgent [205] employs an LLM-based sensory memory module to encode raw observations while filtering noise and irrelevant content. Extending

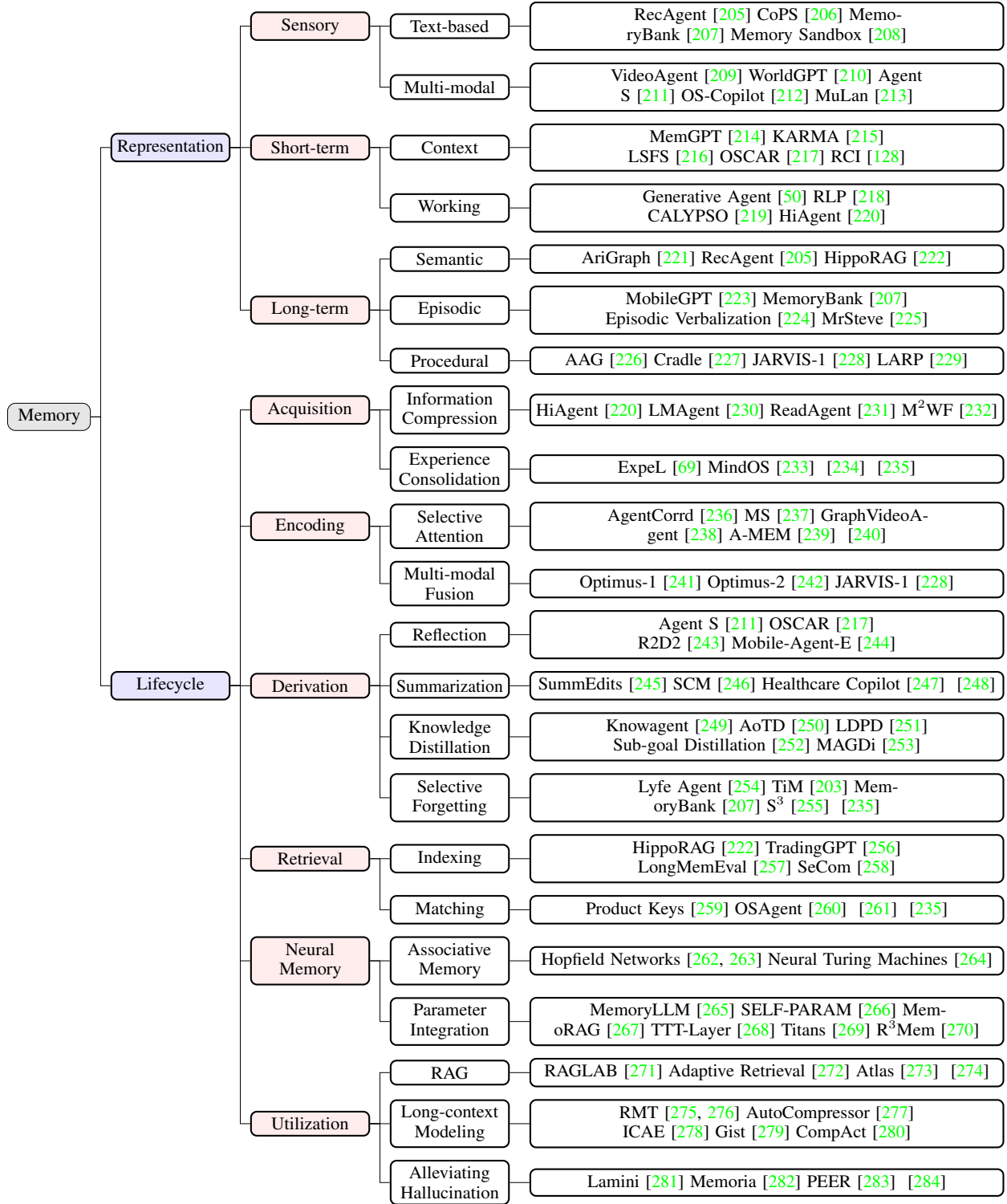


Figure 3.6: Tree diagram of the memory module in intelligent agents.

beyond text-based perception, multimodal sensory memory systems such as Jarvis-1 [228], VideoAgent [209], and WorldGPT [210] integrate multimodal foundation models to process diverse modality inputs.



Next, attentional selection extracts crucial information from the encoded sensory data. This process, guided by an attention mechanism, is represented as:

$$\alpha_t = \text{Attention}(\phi(o_t), c_t) \quad (3.2)$$

where  $\phi(o_t)$  is the encoded input, and  $c_t$  denotes contextual information influencing attention. For example, RecAgent [205] employs an attention mechanism with an importance scoring system that assigns relevance scores to compressed observations, prioritizing critical inputs such as item-specific interactions while de-emphasizing less significant actions. This helps extract high-priority information for memory retention.

Finally, transient retention temporarily stores the selected sensory information as sensory memory:

$$M_{\text{sensory}} = \{\alpha_t \mid t \in [t - \tau, t]\} \quad (3.3)$$

Several strategies have been implemented to manage the time window. For instance, RecAgent [205] models retention by associating each observation with the timestamp corresponding to the start of a simulation round in the user behavior simulation environment, represented as a triplet (observation, importance score, timestamp). Similarly, CoPS [206] employs a fixed-size sensory memory pool as a time window, which consists of user search requests for personalized search, facilitating “re-finding” behavior. When a new query is received, the system first checks the sensory memory for relevant matches. If a match is found, the query is classified as a re-finding instance, enabling a rapid sensory response.

### 3.3.2 Short-Term Memory

Short-term memory in cognition-inspired intelligent agents serves as a transient and dynamic workspace that bridges sensory memory and long-term memory. It is essential for storing and processing task-relevant information and recent interaction sequences, supporting real-time decision-making and adaptive behavior. Inspired by human short-term and working memory, it temporarily retains information to facilitate complex cognitive tasks, ensuring continuity and coherence in the agent’s operations.

Short-term memory in intelligent agents can be categorized into *context memory* and *working memory*. On the one hand, context memory treats the context window as the short-term memory of LLMs. For example, MemGPT [214], inspired by hierarchical memory systems in operating systems, manages different storage tiers to extend context beyond the LLM’s inherent limitations. [290] introduces a neurosymbolic context memory that enhances LLMs by enabling symbolic rule grounding and LLM-based rule application.

On the other hand, working memory involves fetching and integrating relevant external knowledge to hold essential information during an agent’s operation. Generative Agent [50] employs short-term memory to retain situational context, facilitating context-sensitive decision-making. Reflexion [48] utilizes a sliding window mechanism to capture and summarize recent feedback, balancing detailed immediate experiences with high-level abstractions for enhanced adaptability. RLP [218] maintains conversational states for speakers and listeners, using them as short-term memory prompts to support dialogue understanding and generation.

For interactive and creative game scenarios, CALYPSO [219] assists Dungeon Masters in storytelling for Dungeons & Dragons by constructing short-term memory from scene descriptions, monster details, and narrative summaries, enabling adaptive storytelling and dynamic engagement. Similarly, Agent S [211] and Synapse [291], designed for GUI-based autonomous computer interaction, define their short-term memory as task trajectories, including actions such as button clicks and text inputs. This formulation supports behavioral cloning and enhances adaptation in novel GUI navigation tasks.

In robotics applications, SayPlan [292] leverages scene graphs and environmental feedback as short-term memory to guide planning and execution in scalable robotic environments. KARMA [215] engages short-term working memory with an effective and adaptive memory replacement mechanism to dynamically record changes in objects’ positions and states. LLM-Planner [293] iteratively updates short-term memory with environmental observation to prompt an LLM for dynamic planning.

### 3.3.3 Long-Term Memory

Long-term memory in cognition-inspired intelligent agents enables the retention and retrieval of information over extended periods, allowing agents to generalize knowledge and adapt to new contexts effectively. Unlike sensory and short-term memory, which handle transient or immediate data, long-term memory supports cumulative learning and cross-task adaptability. It mirrors human long-term memory by incorporating explicit and implicit components, facilitating richer contextual understanding and intuitive behavior.

On the one hand, *explicit memory* involves intentional recollection, analogous to declarative memory in humans. It consists of *semantic memory*, which stores general knowledge such as facts and concepts, and *episodic memory*,



which records specific events and interaction histories. Semantic memory in intelligent agents can be preloaded from domain knowledge bases or dynamically acquired through interactions. For example, in environments like TextWorld, semantic memory captures structured facts, such as “*Recipe – contains – Tuna*” or “*Recipe – is on – Table*”. Episodic memory, in contrast, logs situational context and sequential actions, such as “go from kitchen to living room, then to garden”. Integrating semantic and episodic memory allows agents to retain static and contextual information, enabling human-like adaptability and context-aware responses.

On the other hand, *implicit memory* shapes agent behavior through *procedural memory* and *priming*. Procedural memory enables agents to perform repetitive tasks efficiently by recalling specific skills and reusable plans. For example, it automates routine tasks without requiring explicit instructions, improving task execution efficiency. Priming, meanwhile, captures state changes and corresponding responses, allowing agents to adapt to similar contexts quickly. Priming enhances fluidity and context-sensitive decision-making by directly matching observations to or continuously chaining actions. Implicit memory, shaped by interactions with cognitive modules, enables rapid adaptation, often after minimal exposure to new stimuli.

Most intelligent agents implement both semantic and episodic memory within their memory modules. For instance, Agent S [211], designed for GUI automation tasks, incorporates semantic memory to store online web knowledge in natural language form, while episodic memory captures high-level, step-by-step task experiences. Similarly, AriGraph [221], targeting embodied simulation tasks, encodes semantic environment knowledge using a fact graph and logs episodic navigation history through an event graph. In AI companion systems like MemoryBank [207] for SiliconFriend, semantic memory constructs user portraits in natural language, while episodic memory retains interaction histories, enhancing personalized and context-aware behavior.

For implementing implicit memory, current agent systems primarily adopt model-friendly memory formats, such as key-value pair storage, executable code, or reusable routines. For example, AAG [226] defines and generalizes procedures through analogy, mapping knowledge from one situation (base) to another (target). This structure can be represented as a linear directed chain graph, where the input serves as the root, the output as the leaf node, and each intermediate step as a node in the chain. Similarly, Cradle [227] and Jarvis-1 [228] implement procedural memory by storing and retrieving skills in code form, which can be either learned from scratch or pre-defined. Once curated, skills can be added, updated, or composed within memory. The most relevant skills for a given task and context are then retrieved to support action planning.

## 3.4 The Memory Lifecycle

In this section, we introduce the lifecycle of memory in AI agents, as depicted in Figure 3.7. The lifecycle comprises a dual process of retention and retrieval. Retention includes acquisition, encoding, and derivation, while retrieval involves memory matching, neural memory networks, and memory utilization.

### 3.4.1 Memory Acquisition

*Memory Acquisition* is the foundational process by which intelligent agents take in raw perceptual information from their environment. This initial step is crucial for subsequent learning, adaptation, and decision-making [305]. A primary challenge in acquisition is the sheer volume and complexity of environmental inputs. Agents are constantly bombarded with visual, auditory, textual, and other forms of data, much of which is redundant or irrelevant to the agent’s goals. Therefore, a core aspect of memory acquisition is not simply capturing data, but also initiating a preliminary filtering process. This filtering leverages two primary mechanisms: initial *information compression* and *experience consolidation*.

At this early stage, information compression involves rudimentary techniques to reduce data dimensionality. This might include downsampling images, extracting key phrases from text using simple heuristics, or identifying significant changes in audio streams [306]. The goal is rapid, lossy compression to prioritize potentially relevant information. For example, LMAgent [230] prompts the LLM to perform information compression, reducing irrelevant and unimportant content when constructing sensory memory to enhance operational efficiency. Meanwhile, ReadAgent [231] and GraphRead [307] respectively employ different strategies for compressing long text, i.e., episode pagination and graph-based structuring, to maximize information retention while ensuring efficiency.

On the other hand, experience consolidation, even at the acquisition phase, plays a role. The agent doesn’t yet have a rich memory, but it can begin to apply previously learned, very general rules or biases. For example, if the agent has a pre-existing bias towards moving objects, it might prioritize visual data containing motion, even before full encoding [308]. To enhance the dynamic consolidation of memory-based experiences, [235] define metrics such as contextual relevance and recall frequency to determine whether to update long-term memory in a vector database.

Method	Domain	Memory Representation			Memory Lifecycle				
		Sensory	Short-term	Long-term	Acquisition	Encoding	Derivation	Retrieval	Utilization
Synapse [291]	GUI	Multi-modal	Context	Episodic, Procedural	User demo.	-	Hierarch. Decomp.	-	-
Agent S [211]	GUI	Multi-modal	Context, Working	Semantic, Episodic	Info. Compress.	Contrastive Learn.	Select. Forget.	Indexing	Long-context
Automanual [108]	GUI	Multi-modal	Context	Procedural, Episodic	User Demo.	Hierarch. Parse	Goal Decomp.	Task Search	Subgoal Exec.
AutoGuide [294]	GUI	Multi-modal	Context	-	Screen Capture	-	Action Plan	-	Action Exec.
Agent-Pro [295]	GUI	Multi-modal	Context	-	Screen Capture	-	Hierarch. Decomp.	-	Action Exec.
MemGPT [214]	Document	Text	Context, Working	-	External Data	-	-	Paging, Func. call	Doc. interact.
SeeAct [296]	Web	Multi-modal	Context	-	Screen Capture	-	Action Plan	-	Web Interact.
AutoWebGLM [297]	Web	Text	Context	-	HTML Parse	HTML Embed	HTML Analysis	-	Web Interact.
SteP [298]	Web	Text	Context	Task-spec.	HTML Parse	HTML Embed	HTML Analysis	Element Rank	Web Interact.
AWM [299]	Web	Text	-	Procedural	Workflow Extract.	Action Summ.	-	Sim. lookup	Workflow exec.
AriGraph [221]	TextWorld	Text	-	Semantic, Episodic	Env. Observ.	Knowl. Graph	Graph Traversal	Assoc. Retrieval	Action plan.
MemoryBank [207]	Dialogue	Text	-	Episodic	Dialogue Record	-	-	Chron. order	Resp. gen.
PromptAgent [300]	General	Text	Context	-	Prompting	-	Prompt Refine.	Content-based	Prompt Exec.
ECL [301]	Embodiment	Multi-modal	Context	Episodic	Obs. Recording	Contrast. Learn.	Exper. Summ.	Sim. & Recency	Policy Learn.
LEO [302]	Embodiment	Multi-modal	Working	Long-Horizon Rep.	Observation	Spatial-Temp. Learn.	Goal-Cond. Policy	Hierarch. Plan	Long-Horizon Exec.
IER [303]	Embodiment	Multi-modal	Context	Episodic	Env. Interact.	Multi-modal Embed	Iter. Refine.	Sim. Match	Action Plan.
Voyager [47]	Embodiment	Text	Working	Procedural	Auto. Curriculum	Skill Library	Iter. Prompt.	-	Skill Exec.
A3T [49]	Embodiment, Robotics	Text	Context	-	Task Decomp.	Token. & Embed.	Action Planning	-	Action select.
STARLING [304]	Robotics	Multi-modal	Context	Procedural	Demo.	Traj. Encode	Skill Refine.	Sim. & Context	Skill Exec.

Table 3.1: Summary of the memory module in various agents. Refer to Figure 3.6 for abbreviations.

Expel [69] constructs an experience pool to collect and extract insights from training tasks, facilitating generalization to unseen tasks. More recently, MindOS [233] proposed a working memory-centric central processing module for building autonomous AI agents, where working memory consolidates task-relevant experiences into structured thoughts for guiding future decisions and actions.

These two mechanisms work in concert with preliminary LLM input. To address the initial challenges, several mechanisms have to be deployed. Agents must be equipped with mechanisms to assess the potential relevance of incoming information rapidly. This preliminary filtering prevents cognitive overload. The acquisition phase also benefits from LLM.

### 3.4.2 Memory Encoding

Memory encoding builds upon acquisition by transforming the filtered perceptual information into internal representations suitable for storage and later use. A key aspect of encoding is selective filtering. This selective attention mimics human cognitive processes [309]. The inherent challenges of encoding stem from the complexity, high dimensionality, and often noisy nature of raw perceptual data. Effective encoding requires advanced mechanisms to identify key

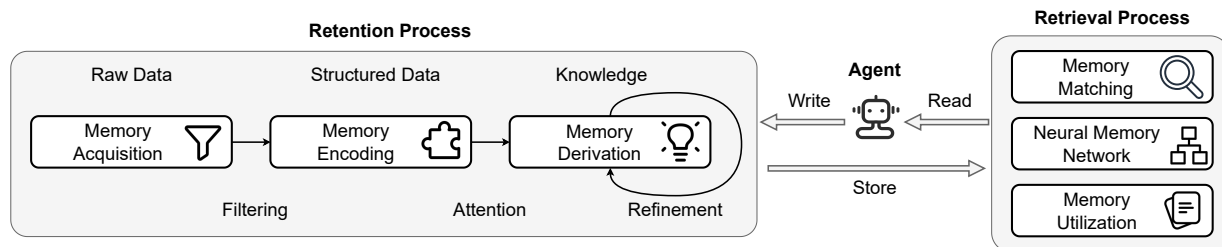


Figure 3.7: Illustration of the memory lifecycle. The memory retention process involves three sequential steps—memory acquisition, encoding, and derivation, while the memory retrieval process encompasses several independent applications, including matching (vector search), neural memory networks, and memory utilization (for long-context modeling and hallucination mitigation).

features, compress them compactly, and integrate information from multiple modalities. Modern approaches address these challenges by leveraging *selective attention* and *multi-modal fusion*.

Selective Attention mechanisms, inspired by human cognition, allow the agent to dynamically focus computational resources on the most relevant parts of the input. This might involve attending to specific regions of an image, keywords in a text, or particular frequencies in an audio signal. Different attention mechanisms can be used depending on the modality and task. For example, as the candidate memory dynamically expands, MS [237] employs an LLM-based scorer to selectively retain the top-scoring half, creating a more compact shared memory across multiple agent systems. In other modalities, GraphVideoAgent [238] utilizes graph-based memory to enable selective and multi-turn video scene understanding, enhancing question-answering performance. In robot control, [240] implements selective attention as a filtering mechanism to extract task-relevant objects from the set of all perceived objects on the table.

Multi-modal Fusion [310] is essential for integrating information from different sensory inputs (e.g., combining visual and auditory data to understand a scene). This involves creating a unified representation space where features from different modalities are aligned. Cross-modal encoders and contrastive learning techniques are often used to achieve this fusion. For example, JARVIS-1 [228] uses the general-domain video-language model CLIP [51] to compute alignment within a multimodal key-value memory, where the key comprises elements such as task, plan, and visual observations, and the value is a text-based representation of successfully executed plans. Furthermore, Optimus-1 [241] refines memory representation and optimizes the multimodal encoder by leveraging MineCLIP [311], a domain-specific video-language model pre-trained on Minecraft gameplay, to align and fuse filtered video streams with textual instructions and plans, encoding the agent’s multimodal experiences into an abstracted memory pool. This integrated representation enhances information retrieval and reasoning across modalities and acts as another filter, reinforcing consistent data. LLMs’ semantic understanding is utilized to extract relevant features efficiently.

### 3.4.3 Memory Derivation

Memory derivation focuses on extracting meaningful knowledge and insights from the acquired and encoded memories. This process goes beyond simple storage. This stage is essential for enhancing the agent’s learning capabilities. The goal is to continuously optimize the structure and content of the agent’s memory. A significant challenge in derivation is the dynamic evaluation of information value. Strategies to address these challenges include *reflection*, *summarization*, *knowledge distillation*, and *selective forgetting*.

Reflection involves an agent actively analyzing its memories to identify patterns, relationships, and potential inconsistencies. It can be triggered by specific events (e.g., an unexpected outcome) or occur periodically as a background process. This process may include comparing memories, reasoning about causal relationships, and generating hypotheses [300]. ExpeL [69] leverages reflection to collect past experiences for generalization to unseen tasks and to support trial-and-error reattempts following failures. R2D2 [243] models memory as a replay buffer and applies reflection to refine it by correcting failed execution trajectories in web agents. These corrected trajectories are then combined with successful ones to construct reflective memory, which serves as a reference for future decision-making.

Summarization aims to produce concise representations of larger bodies of information while preserving their most essential content. This can include extracting key sentences from a document, generating abstractive summaries of conversations, or condensing sequences of events. Summarization techniques range from simple extractive methods to advanced abstractive approaches powered by large language models (LLMs) [245, 312, 246]. For example, [248] introduces a recursive summarization strategy over dialogue history and prior memory to support long-term dialogue memory derivation. Building on this, Healthcare Copilot [247] maintains concise memory by transforming conversation

memory, representing the full ongoing medical consultation, into history memory that retains only key information relevant to the patient’s medical history.

Knowledge distillation [313] enables agents to transfer knowledge from larger, more complex models (or ensembles) to smaller, more efficient ones. This is particularly important for resource-constrained agents and for enhancing generalization. Distillation can also involve consolidating knowledge from multiple specialized models into a single, general-purpose model. For example, AoTD [250] distills textual chains of thought from execution traces of subtasks into a Video-LLM to enhance multi-step reasoning performance in video question answering tasks. LDPD [251] transfers decision-making outcomes from teacher agents (i.e., expert buffers) to student agents, optimizing the student’s policy to align with the teacher’s. In multi-agent systems, MAGDi [253] distills the reasoning interactions among multiple LLMs into smaller models by structurally representing multi-round interactions as graphs, thereby improving the reasoning capabilities of smaller LLMs.

Selective forgetting [314] is the crucial process of removing or down-weighting memories that are deemed irrelevant, redundant, or outdated. This is essential for maintaining memory efficiency and preventing cognitive overload. Forgetting mechanisms can be based on time (older memories are more likely to be forgotten) [247], usage frequency (infrequently accessed memories are more likely forgotten) [203], and relevance to the current task or context [255]. In more fine-grained forgetting mechanisms, MemoryBank [207] applies the Ebbinghaus Forgetting Curve to quantify the forgetting rate, accounting for both time decay and the spacing effect, i.e., the principle that relearning information is easier than learning it for the first time. In contrast, Lyfe Agent [254] adopts a hierarchical summarize-and-forget strategy: it first clusters related memories, refines them into concise summaries, and then removes older memories that are highly similar to newer ones. This approach enables efficient, low-cost memory updates for real-time social interactions.

#### 3.4.4 Memory Retrieval and Matching

Memory retrieval is a process that emulates the human ability to recall relevant knowledge and experiences to solve problems. The goal is to efficiently and accurately extract the most pertinent memory fragments from a large and diverse memory pool, encompassing sensory, short-term, and long-term memory, to inform the agent’s decisions, planning, and actions. Just as humans rely on past experiences to navigate complex situations, agents require a sophisticated memory retrieval mechanism to handle a wide range of tasks effectively.

However, achieving this goal presents several significant challenges. First, the agent’s memory repository is often heterogeneous, comprising various forms of memory such as natural language descriptions, structured knowledge graphs, and state-action-reward sequences. These memories differ fundamentally in their data structures, representations, and levels of semantic granularity, posing a challenge for unified retrieval. Second, the retrieved memory fragments must be highly relevant to the current context, including the agent’s state, task goals, and environmental observations. Simple keyword matching falls short of capturing the deeper semantic relationships required for meaningful retrieval. Developing a context-aware semantic matching mechanism that can dynamically adjust the retrieval strategy based on the current situation is therefore paramount. Third, the real-time nature of agent interaction with the environment necessitates efficient memory retrieval to support rapid decision-making and action [315]. This demand for efficiency is further compounded by the limitations of the agent’s computational resources. Finally, the agent’s memory is not static but constantly evolving as new experiences, knowledge, and skills are acquired. Ensuring memories’ timeliness, reliability, and relevance while avoiding the interference of outdated or erroneous information is a continuous challenge.

A comprehensive approach can address these challenges, encompassing four key components. Firstly, a foundational step involves constructing a unified memory representation and indexing scheme. This aims to bridge the representational gap between different memory types by embedding them into a common vector space. Pre-trained language models like BERT or Sentence-BERT [316] can be leveraged to transform text-based memories into semantic vectors, while graph neural networks (GNNs) can learn vector representations for structured memories like knowledge graphs, capturing both node and edge relationships [317]. To facilitate efficient retrieval, a multi-layered hybrid indexing structure is essential. This integrates techniques like inverted indexes for keyword matching, vector indexes like Faiss [318] or Annoy [319] for similarity search, and graph indexes for structural queries [320], thus supporting diverse query needs.

Secondly, perhaps most critically, the system must develop context-aware semantic similarity computation. This allows the retrieval process to understand and utilize the current context, such as the agent’s state, goals, and observations, enabling a deeper semantic match beyond keyword overlap. This involves encoding the contextual information into vector representations and effectively fusing them with memory vectors. The attention mechanism plays a crucial role here, dynamically calculating the relevance between context and memory vectors and assigning different weights to memory fragments based on their contextual relevance [261]. This emphasizes memories that are more pertinent to the current situation.

Thirdly, integrating memory retrieval with the agent’s task execution necessitates a task-oriented sequence decision and dynamic routing mechanism. This leverages the structural information of tasks to guide memory retrieval and utilization, enabling complex task decomposition, planning, and dynamic adjustments. By constructing a task dependency graph, the agent can topologically sort subtasks to determine execution order. During execution, each subtask’s goal serves as context for memory retrieval, extracting relevant knowledge and experience. Moreover, the agent must adapt to environmental feedback and task progress, dynamically adjusting the execution plan. Each decision point involves re-retrieving memories based on the current state and goal to select the optimal action and handle unexpected situations. This aspect also emphasizes how agents can leverage their skill memory to solve problems, including skill distillation, combination, and innovation. Pattern recognition allows for summarising general problem-solving steps, while structured knowledge organization arranges skills into a retrievable format. Agents can further distill generalized skills from specific ones, combine multiple skills to address complex challenges, and even innovate new skill combinations. These processes depend fundamentally on an efficient memory retrieval system that can identify appropriate skills or skill combinations based on task requirements.

Finally, a robust memory management mechanism is crucial for maintaining the memory pool’s timeliness, relevance, and efficiency. This mechanism should incorporate a forgetting and updating strategy, mirroring human forgetting mechanisms [321]. This might involve regularly purging outdated, redundant, or infrequently used memories based on time-based decay (weakening memory strength over time) and frequency-based decay (purging low-frequency memories). Simultaneously, when a memory fragment relevant to the current task is retrieved, its timestamp and access frequency are updated, increasing its importance and ensuring dynamic memory updates. Through these concerted efforts, LLM Agents can be equipped with a powerful, flexible, and context-aware memory retrieval and matching system, enabling them to effectively utilize their accumulated knowledge, support complex decision-making, and exhibit more intelligent behavior.

### 3.4.5 Neural Memory Networks

Neural Memory Networks represent a fascinating frontier in AI research. They aim to integrate memory seamlessly into the fabric of neural networks. This approach departs from traditional memory architectures by encoding memories directly within the network’s weights or activations, transforming the network into a dynamic, read-write memory storage medium. This tight integration promises significant advancements in efficiency and the utilization of stored information. However, realizing this vision presents several formidable challenges.

A primary concern is balancing memory capacity with stability. Encoding a vast amount of information within the finite parameters of a neural network while maintaining long-term stability poses a major hurdle. The network must be able to store a multitude of memories without succumbing to catastrophic forgetting or confusion between similar memories. Equally crucial is the development of effective mechanisms for memory read-write operations. The network needs to reliably write new information, update existing memories, and accurately retrieve stored information on demand, all while maintaining computational efficiency. Beyond simply storing memories, the ultimate goal is to endow neural networks with the ability to generalize from and reason with the information they store. This would empower them to perform higher-order cognitive functions beyond rote memorization, allowing for insightful connections and inferences based on past experiences. Several approaches are being explored to address these challenges, notably through *associative memory* and *parameter integration*.

On the one hand, associative memory, inspired by the interconnectedness of neurons in the brain, offers a promising avenue. Models like Hopfield networks [262, 263], leveraging energy functions, and Bidirectional Associative Memories (BAMs) [322], supporting hetero-associative recall, provide mechanisms for encoding and retrieving patterns based on the weights between neurons. Besides, Neural Turing Machines (NTMs) [264] and Memory-Augmented Neural Network (MANNs) [323, 324, 275, 265] augment neural networks with external memory modules, employing attention and summary mechanisms to interact with these memories.

On the other hand, parameter integration represents another key research direction, aiming to encode memory directly within a network’s weights. This facilitates the seamless integration of world knowledge and accumulated experience into the operational behavior of intelligent AI agents. For example, some prior works modify model parameters to enable continual learning by updating [325, 326, 327] or forgetting specific knowledge [328]. Other studies treat LLMs as standalone memory modules, incorporating world knowledge into their parameters during pre-training [329], post-training [330], and online deployment [331]. For instance, MemoryLLM [265] introduces memory tokens, while SELF-PARAM [266] leverages knowledge distillation to embed world knowledge and past AI agent experiences into model parameters. This approach is further augmented in the M+ model [332] with a long-term memory mechanism and a co-trained retriever, enhancing its ability to generalize to longer history memorization. Additionally, [333] employs encoded memory to facilitate further reasoning, thereby improving the generalization of stored knowledge. More recently, MemoRAG [267] and R<sup>3</sup>Mem [270] have been proposed to not only encode memory but also enable



reliable retrieval from neural memory networks, unifying the dual processes of memory storage and retrieval within a single model. This advancement contributes to the development of next-generation generative-based retrieval systems, which support lifelong AI applications. Furthermore, Titans [269] have been introduced to memorize test-time data points through meta-learning, enabling more efficient test-time cross-task generalization.

Future research will continue to focus on creating larger capacity and more stable neural memory models. Concurrently, developing more efficient and flexible memory read-write mechanisms will be crucial. A critical area of investigation will involve applying these memory-augmented networks to complex cognitive tasks, pushing the boundaries of what AI can achieve. Progress in this domain will unlock new possibilities for building intelligent agents that can learn, remember, and reason in a manner that is increasingly reminiscent of human cognition.

### 3.4.6 Memory Utilization

A critical aspect of agent design lies in memory utilization, which focuses on maximizing the value of stored memory segments for the current task. The core objective is to apply these memories effectively and appropriately to enhance reasoning, decision-making, planning, and action generation, ultimately boosting the agent’s performance and efficiency while avoiding the pitfalls of irrelevant or incorrect memory interference. Achieving this, however, presents several challenges.

One primary challenge is balancing the vastness of the memory store with its effective utilization. Agents must navigate a potential information overload, ensuring that relevant memories are fully leveraged without overwhelming the system. Another hurdle is the need for abstraction and generalization. Agents need to distill specific memory segments into more general knowledge and apply this knowledge to new and varied situations. Furthermore, the issue of hallucinations and incorrect memories within the LLM requires careful consideration. Preventing the generation of content that contradicts or misrepresents stored information is crucial, as is the ability to identify and rectify erroneous information that may reside within the memory store itself.

To address these challenges, several strategies are employed. *Retrieval-augmented generation (RAG)* [334] combines retrieval and generation models to enhance the LLM’s capabilities by drawing upon external knowledge sources. Unlike the methods mentioned in memory retrieval and matching, RAG focuses on integrating retrieved information into the generation process itself. When prompted, the agent retrieves relevant memory segments and incorporates them into the context provided by the generation model. This contextual enrichment guides the model towards more factual and informative outputs. For instance, when responding to a user’s query, the agent can first retrieve related entries from its knowledge base and then generate an answer based on this retrieved information, thus grounding the response in established knowledge. More recently, some studies have integrated memory modules with RAG, incorporating self-reflection [274] and adaptive retrieval mechanisms [272] to enhance both generation reliability and efficiency. For example, Atlas [273] leverages causal mediation analysis, while [284] employs consistency-based hallucination detection to determine whether the model already possesses the necessary knowledge—allowing for direct generation—or whether retrieval is required, in which case the model first retrieves relevant information before generating a response. In a unified framework, RAGLAB [271] offers a comprehensive ecosystem for evaluating and analyzing mainstream RAG algorithms. HippoRAG [222] employs a strategy inspired by the hippocampal indexing theory of human memory to create a KG-based index for memory and use Personalized PageRank for memory retrieval.

Furthermore, *long-context modeling* plays a vital role in managing extensive memory stores. This approach enhances the LLM’s ability to process long sequences and large-scale memories, allowing for a deeper understanding and utilization of long-range dependencies. By employing Transformer model variants like Transformer-XL [324] and Longformer [335], or through hierarchical and recursive processing techniques, such as recurrent memory transformer (RMT) [275, 276], agents can expand their context window. This enables them to handle significantly more extensive memory stores and reason and make decisions within a much broader context. For example, agents can maintain a longer memory span when processing extensive documents or engaging in prolonged conversations. Additionally, some studies leverage memory to compress long contexts, enabling more effective long-context modeling. For example, AutoCompressor [277] introduces summary vectors as memory to transfer information from previous context windows into the current window, facilitating long-context understanding. Similarly, the in-context autoencoder (ICAE) [278] generates memory slots that accurately and comprehensively represent the original context, while LLMLingua [336, 337], Gist [279], and CompAct [280] further optimize long-prompt compression to reduce input context length.

Finally, *hallucination mitigation* strategies are essential for ensuring the reliability of generated outputs. These strategies aim to minimize the LLM’s tendency to produce factually incorrect or nonsensical content. One approach is implementing fact-checking mechanisms [338], verifying generated content against established knowledge or memory stores. Another involves uncertainty estimation [339, 340], where the model evaluates the confidence level of its generated content and flags or filters out low-confidence outputs. Additionally, knowledge-based decoding strategies can



be employed during the generation phase, introducing constraints that guide the model towards more factually accurate content. These techniques collectively contribute to generating more trustworthy outputs and aligned with the agent’s established knowledge base. Recent research has introduced expert memory subnetworks, such as PEER [283] and Lamini Memory Tuning [281], which specialize in memorizing specific types of information, including world knowledge and AI agents’ past experiences. These subnetworks offload memorization to dedicated parameters, reducing the main model’s propensity to hallucinate. By implementing these memory utilization strategies, agents can become more capable, accurate, and reliable. They can successfully leverage their memory stores to achieve superior performance across complex tasks.

### 3.5 Summary and Discussion

The development of truly intelligent agents depends not just on robust memory systems, but also on their seamless integration with other cognitive functions like perception, planning, reasoning, and action selection. Memory is not an isolated module; it is deeply intertwined with these other processes. For example, sensory input is encoded and filtered before storage (as discussed in the sections on memory representation and lifecycle), highlighting the interplay between perception and memory. Long-term memory, especially procedural memory, directly informs action selection through learned skills and routines. Retrieval mechanisms, like context-aware semantic similarity computation, are crucial for planning, allowing agents to access relevant past experiences. This interplay extends to the concept of a “world model.”

Central to intelligent agents is their ability to build and utilize internal world models. These models, representing an agent’s understanding of its environment, enable simulation, reasoning about consequences, and prediction. Robust world models are crucial for higher-level cognition, planning, and human-like intelligence. A world model is, in essence, a highly structured, often predictive, form of long-term memory. Memory provides the raw material—knowledge and experiences—for constructing the world model, while the world model, in turn, acts as an organizing framework, influencing how new memories are encoded, consolidated, and retrieved. For instance, a well-developed world model might prioritize storing surprising events, as these indicate gaps in the agent’s understanding.

However, developing effective world models and memory systems presents significant challenges. These include managing the complexity of real-world environments, determining the appropriate level of abstraction (balancing accuracy, complexity, and computational efficiency), and integrating multi-modal information. Learning and updating these models efficiently, avoiding bias, ensuring generalization, and enabling continuous adaptation are also critical. Furthermore, model-based planning requires efficient search algorithms to handle the inherent uncertainty in the model’s predictions.

Future research should focus on enhancing agent memory systems by drawing inspiration from the strengths of human memory, particularly its flexibility, adaptability, and efficiency. While agent memory has advanced considerably, it still lags behind human memory in these key areas. Human memory is remarkably associative, retrieving information from incomplete or noisy cues, and it exhibits a sophisticated form of “forgetting” that involves consolidation and abstraction, prioritizing relevant information and generalizing from experiences. Agent memory, conversely, often relies on precise matching and struggles with ambiguity.

Several promising research directions emerge. Exploring biologically-inspired mechanisms, such as neural memory networks (as discussed earlier), could lead to significant breakthroughs. Another crucial area is developing memory systems that actively “curate” their contents—reflecting on information, identifying inconsistencies, and synthesizing new knowledge. This requires integrating metacognitive capabilities (monitoring and controlling one’s own cognitive processes) into agent architectures. Furthermore, creating more robust and nuanced forms of episodic memory, capturing not just the “what” and “when” but also the “why” and the emotional context of events, is essential for agents that can truly learn from experience and interact with humans naturally.

Overcoming these challenges requires innovative solutions at the intersection of deep learning, reinforcement learning, and cognitive science. Developing more sophisticated and adaptable world models and memory systems—ones that mirror the strengths of human cognition—will pave the way for agents with a deeper understanding of their environment, leading to more intelligent and meaningful interactions.